Jacob Bentzen

May 30st, 2022

Foundations Of Programming: Python

Assignment07

Starting from Scratch: Exception Handling and Pickling

**Introduction**

In assignment 7, I was tasked with creating a new program from scratch to demonstrate how exception handling and pickling are used and work in Python.

**Researching Exception Handling and Pickling:**

Exception Handling:

Here are links for some sites I found that were helpful in understanding exception handling in Python:

<https://python-course.eu/python-tutorial/errors-and-exception-handling.php>

This first site listed did a good job of breaking down the different types of exception handling and providing more technical explanations than were shown in the notes.

<https://www.programiz.com/python-programming/user-defined-exception>

This next site was particularly helpful in clearly showing user defined exceptions and how they should be added to my code.

Pickling:

Here are links for some sites I found that were helpful in understanding pickling in Python:

<https://linuxhint.com/pickle_python/>

This first page was useful in properly implementing a for loop to read all of the data from the pickled file.

<https://www.datacamp.com/tutorial/pickle-python-tutorial>

This page had a very in depth but easy to understand list of definitions that were great to really solidify my understanding of basic pickling concepts.

**Creating the Program:**

Coming up with ideas:

First, it will be important to brainstorm some ideas and create a framework for what I want to make. My initial thought was to create three separate small programs: One showing exception handling, one showing pickling, and one showing both at the same time. However, as I started, I basically ended up creating a program that worked together and just wound up turning it into a larger program like ones that we have worked on before (Fig. 1).

Text

Description automatically generated

*Fig. 1: Header and part of the body of the main code*

Essentially my program takes some of the elements of the home inventory program and instead just creates a list of numbers. By performing different actions on the list of numbers, I was able to show different examples of both exception handling and pickling in a concise manner.

Using Exception Handling:

I made sure to put examples of exception handling wherever it made sense.

The first exceptions that are used in the code are an example of raising custom exceptions (Fig. 2).

Text

Description automatically generated

*Fig. 2: Shows the exception class for when a value entered is not a number.*

This is used in the code used to numbers from the list (Fig 3).

Text

Description automatically generated

*Fig. 3: The code used to append values to the main list*

The exception prevents the user from adding in values that are not 1-9, so they can’t put letters or symbols in the list. It raises the custom error if they attempt to do so.

There is also a value error used if the user tries to remove a value not in the list( Fig. 4).

Text

Description automatically generated

*Fig. 4: Shows a built in python exception in action.*

The value error exception is one built in to Python so I did not need to do anything other than reference it in the except clause.

I also used the general exception to catch any errors that might occur upon reading the file (Fig. 5).

Text

Description automatically generated

*Fig. 5: Using a generalized Python exception.*

This was mainly included in the case that the user attempted to read the file and it had not yet been created, in which case the program *would* throw and an error.

Using Pickling:

Picking was much more straightforward for this assignment as I basically used it in a similar fashion to what was shown in the notes (Fig. 6).

Text

Description automatically generated

*Fig 6: Reading from and writing to the file using pickling*

Using the load function was a bit tricky as it only loads one thing at a time. However, that object can be a whole list, not just a single string. In this case, since the program saves a list to the file, it can be extracted as a list and printed out using a for loop over the list. This was definitely an interesting interaction that I might not have noticed had I neglected to utilize both ends of pickling.

**Testing the Script**

Using PyCharm:

Since the program was written in PyCharm, it is straightforward to run and test the completed program (Fig. 7).

Text

Description automatically generated

*Fig. 7: Testing the program in the PyCharm run window.*

Using the Command Window:

After running the script in PyCharm, I made sure to test it using the Command Window on my computer (Fig. 8).

Text

Description automatically generated

*Fig. 8: Command Window with the Assignment 7 Program completed.*

**Summary**

I was able to complete this program by utilizing my understanding of both pickling and exception handling, as well as loops and data processing.

Additionally, here is my GitHub link:

<https://github.com/bentzj2/ITFnd100-Mod07/blob/main/docs/index.md>